Function Approximation Based Reinforcement Learning for Edge Caching in Massive MIMO Networks

Function Approximation Based Reinforcement Learning for Edge Caching in Massive MIMO Networks

Navneet Garg, Mathini Sellathurai, Vimal Bhatia, Tharmalingam Ratnarajah

Abstract

Caching popular contents in advance is an important technique to achieve low latency and reduced backhaul congestion in future wireless communication systems. In this article, a multi-cell massive multi-input-multi-output system is considered, where locations of base stations are distributed as a Poisson point process. Assuming probabilistic caching, average success probability (ASP) of the system is derived for a known content popularity (CP) profile, which in practice is time-varying and unknown in advance. Further, modeling CP variations across time as a Markov process, reinforcement Q-learning is employed to learn the optimal content placement strategy to optimize the long-term-discounted ASP and average cache refresh rate. In the Q-learning, the number of Q-updates are large and proportional to the number of states and actions. To reduce the space complexity and update requirements towards scalable Q-learning, two novel (linear and non-linear) function approximations-based Q-learning approaches are proposed, where only a constant (4 and 3 respectively) number of variables need updation, irrespective of the number of states and actions. Convergence of these approximation-based approaches are analyzed. Simulations verify that these approaches converge and successfully learn the similar best content placement, which shows the successful applicability and scalability of the proposed approximated Q-learning schemes.

Index Terms

Linear function approximation; massive MIMO; non-linear function approximation; Poisson point process; Q-learning; wireless edge caching.

I. Introduction

With the continuous development of various intelligent devices such as smart vehicles, smart home appliances, mobile devices, and various sized innovative applications such as news updates,

high quality video feeds and software updates, wireless mobile communications has been experiencing an unprecedented surge in traffic with a lot of redundant and repeated information, which limits the capacity of the fronthaul and backhaul links [1], [2]. To reduce the redundant traffic, caching has emerged as an effective solution for reducing the peak data rates by prefetching the most popular contents in the local cache storage of the base stations (BS). In the recent years, caching at the BS is actively feasible due to the reduced cost and size of the memory [3]. In wireless networks such as cache enabled macro-cell networks, heterogeneous networks, D2D networks, etc. [3], for a given set of content library and the respective content popularity (CP) profile, content placement and delivery have been investigated in order to optimize the various performance measures like backhaul latency delay [4], server load [5] and cache miss rate [6], [7]. With the known CP profile, in [6], [7], the content placement in cellular networks is optimized to maximize the cache hit rate, while authors in [8], [9] obtain optimal placement policy to maximize the success probability and area spectral efficiency. On a similar note, the approaches in [10], [11] relies on minimizing cache miss probability to get caching policy. However, in practice, CP profile is not known in advance and needs to be estimated from the past observations of the content requests. Deep learning based prediction are effective; however, require huge training data in [12], [13]. In [14], auto regressive (AR) prediction is used to predict the number of requests in the time series, whereas linear prediction approach is investigated for video segments in [15]. Transfer learning methods are used in [16] by leveraging content correlation and information transfer between time periods. To learn CP independently across contents, online policies are presented for cache-awareness in [17], low complexity video caching in [2], [18], user preference learning in [19], etc.

In the literature [6], [7], [10], considering the network as a whole, geographical caching in the Poisson point process (PPP) network is employed for multi-cell system to maximize cache hit rate with respect to the content placement probabilities (CPPs), which represent availability of contents at the BSs. Similarly, in [8], the area success probability and area spectral efficiency are maximized for CPPs. In these works, PPP has been a useful tool to assess the performance of a given network [20], [21]. Therefore, it is important to understand the caching performance variations with respect to time [22]. Since the CP changes dynamically in both time and space due to randomness of the user requests, placement strategies needs to be updated accordingly. Recently, Q-learning based solutions [23]–[26] provide active caching solutions to dynamically

changing content placements via modeling the popularity profile in different time slots as a Markov process. Therefore, in context of PPP analysis, the timely updation of CPPs for time-varying CPs need to be investigated for future wireless systems such as massive-MIMO.

A. Motivation and Contributions

In this paper, a multi-cell massive-MIMO system is considered, where the locations of both the BS and the users are distributed as homogeneous PPPs. In this system, content requests are characterized using a global CP profile, while cache placements are defined via CPPs. Each BS is assumed to simultaneously communicate with multiple users, which makes the success probability more difficult to analyze as compared to the analysis for the case of single antenna BS with single user in [27]. Towards that, first, we derive the success probability, followed by the average success probability (ASP) as a function of CPs and CPPs. For interference limited system, it is shown that the ASP is independent of the density of BSs, since transmissions from BSs depend on the cached contents. If the density of BSs is increased while keeping caching probability fixed, then both the desired and interference signals get stronger, resulting in minor change in the SINR (signal to interference plus noise ratio) and the ASP. Further, since CP is time varying, CP is modeled as a Markov process and the cache placement problem is formulated in terms of conventional Q-learning framework, where the number of Q-updates are proportional to the number of states and actions, incurring large space time complexity for updation. To reduce the computation and updation requirements of Q-learning and to make it scalable with the content library and sizes of state and action sets, two Q-learning approaches are proposed based on linear and non-linear function approximations. In these approaches, only a few variables needs to be updated instead of whole Q-matrix. Furthermore, the convergence of these proposed approaches are analyzed and verified via simulations. The contributions of the paper can be summarized as follows.

1) ASP analysis: For a PPP based multi-cell multi-user massive MIMO system, the ASP expression is derived using stochastic geometric tools. For interference limited systems, it is found that ASP does not depend on the density of BSs. These observations are verified via simulations.

2) Q-learning framework: For time-varying CPs, the problem of dynamically learning the content placement strategies is formulated in terms of Q-learning framework, where the objective

is to maximize the long-term discounted ASP and cache refresh rate. The drawback of Q-learning is the update requirement of a large number of variables proportional to the number of states and actions, which is not feasible and scalable in practice.

3) Function approximation based Q-learning: To improve the Q-learning, Q-function is approximated such that only a few variables needs to be updated. The linear function approximation requires four variables, while the non-linear one needs three. Moreover, we analyze the convergence of the linear and non-linear approximated approaches, and verify their performances through simulations.

Organization: The paper is organized as follows: Section II describes the system model. In Section III, ASP has been derived. Section IV describes the framework of Q-learning, while Section V presents the proposed Q-learning approaches with function approximations. Simulation results are provided in Section VI. Section VII concludes the paper.

II. System Model

We consider a cache-enabled multi-cell system, where each BS, equipped with an array of large number of antennas M, serves multiple single antenna users. The locations of BSs and users are independently distributed as homogeneous PPPs Φ_{BS} and Φ_u, with the corresponding densities λ_{BS} and λ_u respectively as shown in Figure 1.

A. Caching Model

We consider a time slotted model [26], where the structure of each time slot is depicted in Figure 2. At the beginning of the time slot, the content placement takes place, which is based on the content popularity and cache information in the previous time slot. The next phase pertains to content delivery, where the cached content is delivered as users' requests arrive. Subsequently, in the information exchange phase, each BS shares the local content requests information to a central station or a designated BS, which forwards back the global popularity profile, computed based on simple averaging or weighted averaging.

Each BS is equipped with a cache storage $\mathcal{L}_t$ of L units at time t, which is filled in the placement phase with a subset of the content library $\mathcal{F} = \{1, 2, \ldots, f, \ldots, F\}$. For simplicity, we assume each content has the same size of one unit [19]. In the information exchange phase of time slot t, based on the number of user requests, the revealed popularity profile is denoted by

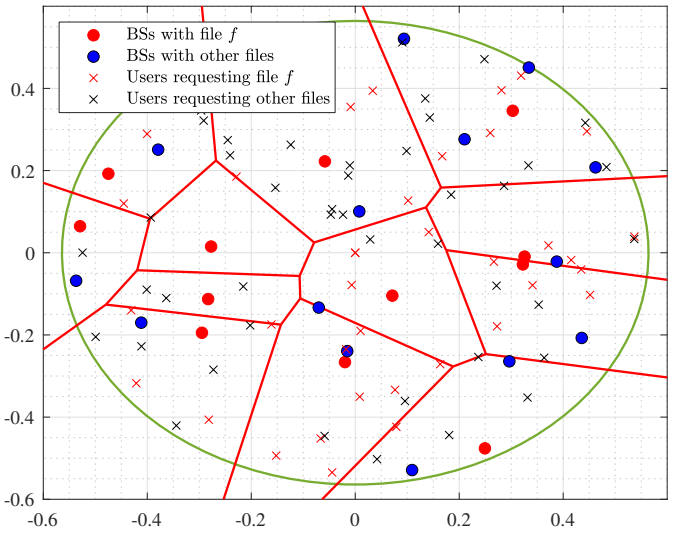


Figure 1. BSs and users distributed as independent homogeneous PPPs. Users color indicate the requesting content. Voronoi region is based on BSs with the requesting file cached (red points).

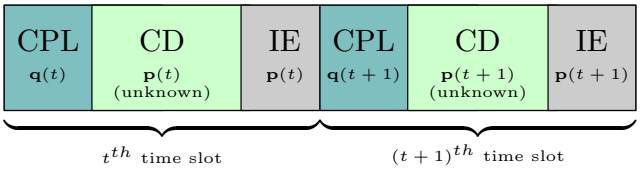


Figure 2. A typical time slot structure in edge caching (CD: Content Delivery, CPL: Content placement, IE: Information Exchange) [26].

$\mathbf{p}_t^T = [p_{1,t}, \ldots, p_{F,t}]$ with $p_{f,t} \geq 0$ and $\sum_{f\in\mathcal{F}} p_{f,t} = 1$. Let $q_{f,t} = \Pr(f \in \mathcal{L}_t)$ denote the cache placement probability of the f^{th} content in the t^{th} time slot, which represents the probability of the f^{th} content being cached at a typical BS using the probabilistic caching as in [6]. These caching probabilities $\mathbf{q}_t^T = [q_{1,t}, \ldots, q_{F,t}]$ satisfy the cache constraint $\sum_{f\in\mathcal{F}} q_{f,t} \leq L$. In the following to derive the ASP, we temporarily drop subscript t and resume in Section IV.

B. Received Signal Model

From the Slivanyak-Mecke theorem, for stationary and homogeneity of PPPs, we consider a typical user at the origin o for evaluating the performance. A typical user connects to the nearest BS who has the desired content. If the requested content is not available in any of the caches

at the BSs, it is considered as a failure and the required file must be fetched from the content server via the backhaul link. Let the k^{th} BS serves K_k users indexed by $\mathcal{K}_k \subseteq \Phi_u$. The received signal at the typical user requesting the f^{th} content from the k^{th} BS can be given as

$$y_{fk} = \bar{\mathbf{h}}_{ok}^T \mathbf{s}_k + \sum_{j \in \Phi_{BS} \setminus \{k\}} \bar{\mathbf{h}}_{oj}^T \mathbf{s}_j + n_{fk}, \tag{1}$$

where $\bar{\mathbf{h}}_{ok}^T = R_{ok}^{-\alpha/2} \mathbf{h}_{ok}^T$; R_{ok} and $\mathbf{h}_{ok}$ are distance and the CSI vector from the k^{th} BS to the typical user; α is the path loss exponent; $\mathbf{s}_k = \mathbf{W}_k \mathbf{x}_k = \sum_{u \in \mathcal{K}_k} \mathbf{w}_{uk} x_{uk}$ is the precoded transmitted signal of the k^{th} BS with $\mathbb{E}\left\{\mathbf{x}_j \mathbf{x}_j^H\right\} = \mathbf{I}_{K_j}$; and $n_{fk} \sim \mathcal{CN}(0, \sigma^2)$ is additive white Gaussian noise. The first term in the above equation corresponds to the desired signal with intra-cell interference, the second term pertains to the inter-cell interference from the other BSs that may have the f^{th} content transmitting to other users.

Transmit Power Constraint: Assuming the total transmit power constraint P_T, we can write $\mathbb{E}\left\{\mathbf{s}_k \mathbf{s}_k^H\right\} = \|\mathbf{W}_k\|_F^2 = \sum_{u \in \mathcal{K}_k} \|\mathbf{w}_{uk}\|_2^2 \leq P_T$. Let p_{uk} denote the per user allocated power. Then, $\|\mathbf{w}_{uk}\|_2^2 = p_{uk}$ and $\sum_{u \in \mathcal{K}_k} p_{uk} \leq P_T$.

Thinning of BSs: Based on the f^{th} content availability, the PPP for BSs can be divided into two PPPs: $\Phi_{BS}(f)$ with density $q_f \lambda_{BS}$, and $\Phi_{BS}^c(f)$ with density $(1 - q_f)\lambda_{BS}$. The BSs with the f^{th} content, indexed by $\Phi_{BS}(f) \setminus \{k\}$, are located at distance $R_{oj} > R_{ok}, \forall j \neq k$ with R_{ok} being the distance of the connected k^{th} BS to the typical user, while the BSs in $\Phi_{BS}^c(f)$ have distance $R_{oj} > 0, \forall j \in \Phi_{BS}$ from the typical user. Thus, the summation in the inter-cell interference can be divided as

$$\Phi_{BS} \setminus \{k\} = \{\Phi_{BS}(f) \setminus \{k\}\} \cup \Phi_{BS}^c(f).$$

SINR Expression: The downlink SINR for the typical user can be obtained as

$$\Gamma_{fk} = \frac{\mathbb{E}\left\{\left|\bar{\mathbf{h}}_{ok}^T \mathbf{w}_{ok}\right|^2\right\}}{\sum_{l \in \mathcal{K}_k \setminus \{o\}} \mathbb{E}\left\{\left|\bar{\mathbf{h}}_{ok}^T \mathbf{w}_{lk}\right|^2\right\} + I_{fk} + I_f^c + \sigma^2}, \tag{2}$$

where in the denominator, the first term, $I_{fk} = \sum_{j \in \Phi_{BS}(f) \setminus \{k\}} \mathbb{E}\left\{\left\|\bar{\mathbf{h}}_{oj}^T \mathbf{W}_j\right\|_2^2\right\}$ and $I_f^c = \sum_{j \in \Phi_{BS}^c(f)} \mathbb{E}\left\{\left\|\bar{\mathbf{h}}_{oj}^T \mathbf{W}_j\right\|\right.$ correspond to the intra-cell interference and the inter-cell interference strengths from the BSs based on the presence of the f^{th} content respectively. The value of these interferences are decided by the BS's transmission strategy.

Maximal Ratio Transmission (MRT): Let $\mathbf{H}_k = [\mathbf{h}_{k,1}, \ldots, \mathbf{h}_{k,K_k}]$ be the concatenated channel vectors for K_k users connected to the k^{th} BS. The presence of massive MIMO BSs allows to utilize the channel hardening effect [28], $\frac{1}{M}\mathbf{H}_k^H\mathbf{H}_k \to \mathbf{I}_{K_k}$, which acts like the expectation operator i.e. $\mathbb{E}\left\{\mathbf{H}_k^H\mathbf{H}_k\right\} = M\mathbf{I}_{K_k}$. Utilizing MRT, the precoder at the k^{th} BS can be written as $\mathbf{W}_k = \frac{1}{\sqrt{M}}\mathbf{H}_k\mathbf{P}_k^{1/2}$, that is, $\mathbf{w}_{uk} = \mathbf{h}_{uk}\sqrt{\frac{p_{uk}}{M}}$, where $\mathbf{P}_k = \mathcal{D}\left(p_{1k}, \ldots, p_{K_k k}\right)$ is a diagonal power allocation matrix such that $\mathbb{E}\left\{\|\mathbf{w}_{ik}\|^2\right\} = p_{ik}$. From (2), the respective downlink SINR can be simplified as[1]

$$\Gamma_{fk}^{MRT} = \frac{R_{ok}^{-\alpha} p_{ok} M}{R_{ok}^{-\alpha}\sum_{l\in\mathcal{K}_k\setminus\{o\}} p_{lk} + P_T\sum_{j\in\Phi_{BS}\setminus\{k\}} R_{oj}^{-\alpha} + \sigma^2} \tag{3}$$

$$= \frac{R_{ok}^{-\alpha}\frac{p_{ok}}{P_T} M}{R_{ok}^{-\alpha}\left(1 - \frac{p_{ok}}{P_T}\right) + \sum_{j\in\Phi_{BS}\setminus\{k\}} R_{oj}^{-\alpha} + \frac{\sigma^2}{P_T}}. \tag{4}$$

Zero Forcing (ZF) based transmission: Optional to MRT, to mitigate the intra-cell interference with ZF precoding, we compute $\mathbf{W}_k = \sqrt{M}\mathbf{H}_k\left(\mathbf{H}_k^T\mathbf{H}_k\right)^{-1}\mathbf{P}_k^{1/2}$ such that $\mathbf{w}_{uk} = \mathbf{H}_k\left(\mathbf{H}_k^T\mathbf{H}_k\right)^{-1}\mathbf{e}_u p_{uk}\sqrt{M}$ and $\mathbb{E}\left\{\mathbf{w}_{ok}^H\mathbf{w}_{ok}\right\} = \mathbb{E}\left\{\mathbf{e}_o^T\left(\mathbf{H}_k^T\mathbf{H}_k\right)^{-1}\mathbf{e}_o\right\} M p_{ok} = p_{ok}$, where $\mathbf{e}_u$ is a u^{th} column of identity matrix. Thus, the resultant SINR can be written as[2]

$$\Gamma_{fk}^{ZF} = \frac{R_{ok}^{-\alpha}\frac{p_{ok}}{P_T} M}{\sum_{j\in\Phi_{BS}\setminus\{k\}} R_{oj}^{-\alpha} + \frac{\sigma^2}{P_T}}. \tag{5}$$

It can be seen that the SINR expression for MRT in (4) is more general than that for ZF in above. Thus, MRT based SINR will be analyzed which can also provide insights about ZF based SINR.

III. Success probability analysis

In this section, considering MRT based SINR expression, success probability is derived, followed by different use cases.

[1]SINR terms for MRT precoding are simplified as
$\mathbb{E}\left\{\left|\bar{\mathbf{h}}_{ok}^T\mathbf{w}_{ok}\right|^2\right\} = R_{ok}^{-\alpha}\frac{p_{ok}}{M}\mathbb{E}\left\{\|\mathbf{h}_{ok}\|^4\right\} = R_{ok}^{-\alpha}\frac{p_{ok}}{M}\left(M^2+M\right) \approx R_{ok}^{-\alpha} p_{ok} M$, $\mathbb{E}\left\{\left|\bar{\mathbf{h}}_{ok}^T\mathbf{w}_{lk}\right|^2\right\} = R_{ok}^{-\alpha}\frac{p_{lk}}{M}\mathbb{E}\left\{\left|\mathbf{h}_{ok}^T\mathbf{h}_{lk}\right|^2\right\} = R_{ok}^{-\alpha} p_{lk}$ and
$\mathbb{E}\left\{\left\|\bar{\mathbf{h}}_{oj}^T\mathbf{W}_j\right\|_2^2\right\} = \sum_{u\in\mathcal{K}_j}\mathbb{E}\left\{\left|\bar{\mathbf{h}}_{oj}^T\mathbf{w}_{uj}\right|^2\right\} = R_{oj}^{-\alpha}\sum_{u\in K_j} p_{uj} = R_{oj}^{-\alpha} P_T$.

[2]For ZF precoding, SINR terms are given as $\mathbb{E}\left\{\left|\bar{\mathbf{h}}_{ok}^T\mathbf{w}_{ok}\right|^2\right\} =$
$R_{ok}^{-\alpha} p_{ok} M\mathbb{E}\left\{\left|\mathbf{h}_{ok}^T\mathbf{H}_k\left(\mathbf{H}_k^T\mathbf{H}_k\right)^{-1}\mathbf{e}_o\right|^2\right\} = R_{ok}^{-\alpha}\frac{p_{ok}}{M}\mathbb{E}\left\{\left|\mathbf{h}_{ok}^T\mathbf{H}_k\mathbf{e}_o\right|^2\right\} = R_{ok}^{-\alpha}\frac{p_{ok}}{M}\mathbb{E}\left\{\|\mathbf{h}_{ok}\|^4\right\} \approx R_{ok}^{-\alpha} p_{ok} M$ and
$\mathbb{E}\left\{\left\|\bar{\mathbf{h}}_{oj}^T\mathbf{W}_j\right\|_2^2\right\} = \sum_{u\in\mathcal{K}_j}\mathbb{E}\left\{\left|\bar{\mathbf{h}}_{oj}^T\mathbf{w}_{uj}\right|^2\right\} = R_{oj}^{-\alpha}\sum_{u\in\mathcal{K}_j} p_{uj} M\mathbb{E}\left\{\left|\mathbf{h}_{oj}^T\mathbf{H}_k\left(\mathbf{H}_k^T\mathbf{H}_k\right)^{-1}\mathbf{e}_l\right|^2\right\} =$
$R_{oj}^{-\alpha}\sum_{u\in\mathcal{K}_j}\frac{p_{uj}}{M}\mathbb{E}\left\{\left|\mathbf{h}_{oj}^T\mathbf{H}_k\mathbf{e}_l\right|^2\right\} = R_{oj}^{-\alpha} P_T$.

Due to concurrent transmissions, the interference at the typical user becomes a dominant factor. From the user's perspective, to maintain a quality of service and to evaluate the caching performance, the success probability measure is considered and is defined as the probability that the achievable rate of a typical user exceeds the rate threshold R_0 for the f^{th} content in a typical time slot as

$$g(q_f) = \mathbb{E}_{k\in\Phi_{BS}}\left\{\Pr\left(W\log_2\left(1+\Gamma_{fk}\right)\geq R_0\right)\right\}, \tag{6}$$

with W being the transmission bandwidth. For the whole content set $\mathcal{F}$, the ASP can be written as

$$P(\mathbf{p},\mathbf{q}) = \mathbb{E}_f\left\{g(q_f)\right\} = \sum_{f\in\mathcal{F}} p_f g(q_f). \tag{7}$$

Since the success probability is difficult to analyze with respect to the PPP of BSs Φ_{BS} and the SINR model in (4), we focus on analyzing another point process with a more tractable SINR model as long as both the point processes have statistically equivalency, which is defined as follows.

Definition 1. Two stochastic point processes Φ_1 and Φ_2 with SINR models Γ_1 and Γ_2 are said to be *statistically equivalent* if the SINR distribution at the typical user is same for both the processes, i.e. $\Pr\left(\Gamma_1 > T\right) = \Pr\left(\Gamma_2 > T\right)$ [29].

Since the evaluation of success probability is not straightforward with Φ_{BS} for the SINR model in (4), we focus on analyzing another PPP for a tractable SINR model as long as both the PPPs are equivalent.

Lemma 2. *The 2D-homogeneous PPP Φ_{BS} and SINR model in (4), is statistically equivalent to another 1D-point process Φ_{eq} with density function $\lambda_{eq}(d) = Cd$, where $C = \frac{2\pi\lambda_{BS}}{\Gamma\left(1+\frac{2}{\alpha}\right)}$. The equivalent SINR model for Φ_{eq} is given as*

$$\Gamma^{MRT}_{fk,eq} = \frac{\xi_{ok} d_{ok}^{-\alpha}\frac{p_{ok}}{P_T}M}{\xi_{ok} d_{ok}^{-\alpha}\left(1-\frac{p_{ok}}{P_T}\right) + \underline{I}_{fk} + \underline{I}^c_f + \bar{\sigma}^2}, \tag{8}$$

where $\underline{I}_{fk} + \underline{I}^c_f = \sum_{j\in\Phi_{BS}\setminus\{k\}}\xi_{oj}d_{oj}^{-\alpha}$ and $\bar{\sigma}^2 = \frac{\sigma^2}{P_T}$.

Proof: Please refer to Appendix-A. ■

The above result transforms the homogeneous PPP into an inhomogeneous PPP, along with the transformation of SINR expression from (4) to (8) with a planar distance path loss, multiplied

by auxiliary random variables, representing the small scale fading. In the following, it will be shown that the above statistical equivalent transformation can significantly simplify the analysis of success probability, owing to exponentially distributed auxiliary random variables ξ_{oj}.

Remark (Equivalent thinning based on caching): Based on the f^{th} content availability, the equivalent point process Φ_{eq} can also be divided into two processes $\Phi_{eq}(f)$ and $\Phi_{eq}^{c}(f)$ with densities $q_f\lambda_{eq}$ and $(1-q_f)\lambda_{eq}$.

Nearest BS Distribution: The cumulative density function of the random distance d_{ok}, which represents the distance to the closest BS having f^{th} file, can be obtained from [30]

$$\Pr(d_{ok} \leq d) = \exp\left(-\int_0^d q_f \lambda_{eq}(z)dz\right), \tag{9}$$

yielding the probability density function of d_{ok} as

$$f_{d_{ok}}(z) = q_f C z \exp\left(-q_f C \frac{z^2}{2}\right). \tag{10}$$

Remark (Inactive probability): For a typical BS, the inactive probability is the probability that it has no users scheduled and is inversely proportional to the relative density of the users per BS [31]. We assume the inactive probability to be negligible, since the relative density is considered to be large.

Based on the statistical equivalent SINR model in (8), the success probability of a typical user is given in the following result.

Theorem 3. *For the MRT transmission, the success probability at a typical user for the f^{th} file can be expressed as*

$$g(q_f) = \mathbb{E}_{d_{ok}\in\Phi_{eq}}\left\{\exp\left(-d_{ok}^2 w_f - T_{ok} d_{ok}^{\alpha}\bar{\sigma}^2\right)\right\}, \tag{11}$$

where $w_f = C\left(q_f A + (1-q_f)B\right)$, $A = \alpha^{-1}T_{ok}^{2/\alpha}I(0)$,
$B = \alpha^{-1}T_{ok}^{2/\alpha}I(T_{ok}^{-1})$, $T_{ok} = \frac{TP_T}{Mp_{ok}} \cdot \frac{1}{1-\frac{T}{M}\left(\frac{P_T}{p_{ok}}-1\right)}$,
$T = 2^{\frac{R_0}{W}} - 1$ *and* $I(x) = \int_x^{\infty} \frac{c^{2/\alpha-1}\mathrm{d}c}{1+c}$.

Proof: Proof is given in Appendix-B. For ZF based transmission, $T_{ok} = \frac{TP_T}{Mp_{ok}}$. ■

Corollary 4. *(Path loss exponent case): When $\alpha = 2$, the value of success probability reduces*

to

$$g(q_f)\Big|_{\alpha=2} = \mathbb{E}_{d_{ok}\in\Phi_{eq}}\left\{\exp\left(-d_{ok}^2\left(w_f + T_{ok}\bar{\sigma}^2\right)\right)\right\} \tag{12}$$

$$= \frac{q_f}{2B + q_f\left(2A - 2B + 1\right) + T_{ok}\bar{\sigma}^2}. \tag{13}$$

Corollary 5. *(Linear approximation): With exponential approximation ($e^{-x} \approx 1-x$), the success probability is reduced to*

$$g(q_f) \approx 1 - \mathbb{E}_{d_{ok}\in\Phi_{eq}}\left\{d_{ok}^2 w_f + T_{ok} d_{ok}^{\alpha}\bar{\sigma}^2\right\} \tag{14}$$

$$= 1 - \frac{2\Gamma(2)}{q_f C} w_f - T_{ok}\Gamma(1+\alpha/2)\left(\frac{2}{q_f C}\right)^{\alpha/2}\bar{\sigma}^2. \tag{15}$$

The above approximation yields the resultant ASP as

$$P(\mathbf{p},\mathbf{q}) \approx 1 - 2A + 2B - 2B\sum_f \frac{p_f}{q_f} - T_{ok}\frac{\Gamma(1+\alpha/2)}{(0.5C)^{\alpha/2}}\bar{\sigma}^2\sum_f p_f q_f^{-\alpha/2}. \tag{16}$$

Corollary 6. *(Interference limited case): For the interference limited case ($\sigma^2 \to 0$), the success probability is computed as*[3]

$$g_0(q_f) = g(q_f)\Big|_{\sigma^2\to 0} = \mathbb{E}_{d_{ok}\in\Phi_{eq}}\left\{\exp\left(-d_{ok}^2 w_f\right)\right\} \tag{17}$$

$$= \frac{q_f}{2B + q_f\left(2A - 2B + 1\right)}. \tag{18}$$

From the above equation, it can be seen that for interference limited regime, the success probability is independent of the density of BSs and users, and dependent on the caching probabilities and the threshold. The reason behind is that the power of both desired and interference signals increase with the increase in the density of BSs, causing minor change in the signal-to-interference ratio for interference limited system and yielding the density-independent ASP. Further, for the f^{th} content, to maximize the success probability, caching probability q_f should be chosen according to the popularity, the threshold and the cache size. For interference limited

[3] $g_0(q_f) = \int_0^\infty \exp\left(-z^2 w_f\right) f_{d_{ok}}(z)\mathrm{d}z = q_f C\int_0^\infty z\exp\left(-z^2 w_f - q_f C\cdot\frac{z^2}{2}\right)dz = \frac{q_f C}{2w_f + q_f C}\int_0^\infty \exp\left(-t\right)dt = \frac{q_f}{2\left(q_f A + (1-q_f)B\right) + q_f}$.

networks, the resulting ASP for the content library $\mathcal{F}$ can be written as

$$P_0(\mathbf{p}, \mathbf{q}) = \mathbb{E}_f \left\{ g_0(q_f) \right\} = \sum_{f \in \mathcal{F}} p_f g_0(q_f). \tag{19}$$

In practice, the popularity is not known in advance. Based on the CP profile in the previous time slots, content is cached ahead of time, when needed i.e. requested by a user. To achieve that, Q-learning approaches are presented in the next sections.

IV. Q-Learning

In this section, we first describe in brief about the dynamics of the Q-learning system, followed by defining the elements of Q-learning. Thereafter, using Bellman's equations, the algorithm for Q-learning is presented.

A. Dynamics

At the first content placement phase of the time slot t as shown in Figure 2, the content is placed in BS caches via caching action based on the information in the previous time slot. After content delivery phase takes place, it is followed by the information exchange phase, where the next state of the system is revealed in terms of the CP $\mathbf{p}_t$. This observation is used to compute the reward, which is used to update the Q-values, and the next state is updated before the end of the time slot t. Note that caching action is taken, before $\mathbf{p}_t$ is observed.

B. System States, Actions and Reward

1) States: At time slot t, the state of the system can be captured in terms of the popularity in the t^{th} time slot, and the content status in the cache. Thus, the state in IE phase of the time slot t is revealed as

$$s_t = (\mathbf{p}_t, \mathbf{q}_t) \in \left\{ \begin{bmatrix} \mathbf{p} \\ \mathbf{q} \end{bmatrix} \in [0,1]^{2F} : \begin{array}{c} \mathbf{p}^T \mathbf{1} = 1 \\ \mathbf{q}^T \mathbf{1} = L \end{array} \right\}, \tag{20}$$

where $\mathbf{1}$ is a column vector of ones, and $\mathbf{q}_t$ denotes the content placement decided based on $\mathbf{p}_{t-1}$. Therefore, the state contains one length history, defining the present state of the system at the end of time slot t.

2) Actions: A caching action is taken at the beginning of the time slot t, and is defined as the content placement $\mathbf{q}_t$,

$$a_t = (\mathbf{q}_t) \in \left\{ \mathbf{q} \in [0,1]^{F\times 1} : \mathbf{q}^T \mathbf{1} = L \right\}. \tag{21}$$

In state s_t, the action a_{t+1} is decided. In other words, the action $\mathbf{q}_{t+1}$ is selected based on the history $(\mathbf{p}_t, \mathbf{p}_{t-1})$, since $\mathbf{q}_t$ (in s_t) was chosen based on $\mathbf{p}_{t-1}$ in the similar way.

3) Transition probability: The probability of transition from states s_t to s_{t+1} via the action a_{t+1} can be defined as

$$\begin{aligned}\Pr(s_{t+1}|s_t, a_{t+1}) &= \Pr\left(\begin{bmatrix} \mathbf{p}_{t+1} \\ \mathbf{q}_{t+1} \end{bmatrix} \Big| \begin{bmatrix} \mathbf{p}_t \\ \mathbf{q}_t \end{bmatrix}, \mathbf{q}_{t+1} \right) \\ &= \Pr(\mathbf{p}_{t+1} \Big| \mathbf{p}_t, \mathbf{q}_t) = \Pr(\mathbf{p}_{t+1} \Big| \mathbf{p}_t),\end{aligned}$$

where the last equality is obtained from the fact that the popularity varies as a Markov process i.e. the popularity at time t depends on that of time $t-1$. Since $\mathbf{q}_t$ is chosen based on $\mathbf{p}_{t-1}$, $\mathbf{p}_{t+1}$ is independent of $\mathbf{q}_t$.

4) Reward: Our objective is to maximize the long term discounted ASP and the cache refresh rate. After observing the popularity $\mathbf{p}_{t+1}$, the reward in the IE phase of the time slot t is defined as a function of ASP and cache refresh rate as

$$r\left(s_t, a_{t+1}, s_{t+1}\right) = P\left(\mathbf{p}_{t+1}, \mathbf{q}_{t+1}\right) - \nu \mathbf{q}_{t+1}^T \left(1 - \mathbf{q}_t\right), \tag{22}$$

where ν is the weight controlling the preferred objective. It can be noted that the next state is random. Thus, the average reward per state can be computed as

$$\begin{aligned} R\left(s_t, a_{t+1}\right) &= \mathbb{E}_s \left\{ r\left(s_t, a_{t+1}, s\right) | s_t, a_{t+1} \right\} \\ &= \mathbb{E}_{\mathbf{p}} \left\{ P\left(\mathbf{p}, \mathbf{q}_{t+1}\right) | \mathbf{p}_t \right\} - \nu \mathbf{q}_{t+1}^T \left(1 - \mathbf{q}_t\right) \\ &= g\left(\mathbf{q}_{t+1}\right)^T \mathbb{E}_{\mathbf{p}} \left\{ \mathbf{p} | \mathbf{p}_t \right\} - \nu \mathbf{q}_{t+1}^T \left(1 - \mathbf{q}_t\right), \end{aligned}$$

where $g^T(\mathbf{q}) = [g(q_1), \ldots, g(q_F)]$. The above reward is composed of two terms. The first term is the ASP, which has been considered as a measure of caching. The better is the content placement, the better is the ASP and the reward. In the second term, $(1 - q_{f,t})$ denotes the not-cached portion of the f^{th} content among BSs, while $q_{f,t+1}$ denotes the portion being cached in the next time slot. Thus, $\left(1 - q_{f,t}\right) q_{f,t+1}$ implies the portion of the f^{th} content being updated, and so, the term $\mathbf{q}_{t+1}^T \left(1 - \mathbf{q}_t\right)$ represents the average cache refresh rate.

In the above, the term $\mathbb{E}_{\mathbf{p}}\left\{\mathbf{p}|\mathbf{p}_t\right\}$ represents the conditional mean estimate of the popularity at time $t+1$, given the previous CP information $\mathbf{p}_t$. In other words, it suggests that the caching problem can also be solved using one-step prediction methods for Markov popularities, when ASP is the only objective. However, cache refresh rate depends on the choice of actions in the previous time slot, and hence cannot be optimized via prediction methods.

C. Value functions

For the above model with the long term expected discounted reward, the state-value function can be written as

$$V\left(\left\{a_t\right\},s\right)=\mathbb{E}_{\{s_t\}}\left[\sum_{t=0}^{\infty}\gamma^t R\left(s_t,a_{t+1}\right)\Big|s_0=s\right] \tag{23}$$

$$=\mathbb{E}_{\{\mathbf{p}_t\}}\left[\sum_{t=0}^{\infty}\gamma^t g\left(\mathbf{q}_{t+1}\right)^T\mathbb{E}_{\mathbf{p}}\left\{\mathbf{p}\Big|\mathbf{p}_t\right\}\Big|\mathbf{p}_0=\mathbf{p}\right]$$

$$+\nu\sum_{t=0}^{\infty}\gamma^t\mathbf{q}_{t+1}^T\left(1-\mathbf{q}_t\right) \tag{24}$$

The above value function can be maximized with respect to the actions $\{a_t\}$ as

$$V^*\left(s\right)=\max_{\{a_t\}}V\left(\left\{a_t\right\},s\right) \tag{25}$$

$$=\max_{\{a_t,t>0\}}\mathbb{E}_{\{s_t\}}\left[R\left(s_0,a_1\right)+\sum_{t=1}^{\infty}\gamma^t R\left(s_t,a_{t+1}\right)\Big|s_0=s\right]$$

$$=\max_{\{a_1,a_t,t>1\}}R\left(s,a_1\right)+\gamma\mathbb{E}_{\{s_t\}}\left[\sum_{t=1}^{\infty}\gamma^{t-1}R\left(s_t,a_{t+1}\right)\Big|s\right]$$

$$=\max_{a_1}R\left(s,a_1\right)+\gamma\max_{\{a_t,t>1\}}$$

$$\mathbb{E}_{s_1|s}\mathbb{E}_{\{s_t,t\geq 1\}}\left[\sum_{t=1}^{\infty}\gamma^{t-1}R\left(s_t,a_{t+1}\right)\Big|s_1,s_0=s\right]$$

$$=\max_{a_1}R\left(s,a_1\right)+\gamma\max_{\{a_t,t>1\}}\mathbb{E}_{s_1|s}\left[V\left(\left\{a_t\right\},s_1\right)\right] \tag{26}$$

$$=\max_{a_1}R\left(s,a_1\right)+\gamma\mathbb{E}_{s_1|s}\left[V^*\left(s_1\right)\right], \tag{27}$$

which is known as the Bellman's equation. Similarly, the optimal state-action Q-function is defined as

$$Q^*\left(s,a\right)=R\left(s,a\right)+\gamma\mathbb{E}_{s_1|s}\left[V^*\left(s_1\right)\right], \tag{28}$$

representing the expected total discounted reward along a trajectory starting at state s, obtained by choosing a as the first action and following the optimal trajectory afterwards. The optimal action set can thus be obtained as

$$a_{t+1}^{*} = \arg\max_{a} Q^{*}\left(s_t, a\right), \tag{29}$$

which is optimal in the sense that $V\left(\left\{a_{t+1}^{*}\right\}, s_t\right) = V^{*}\left(s_t\right)$ and it leads to a mapping π^{*} : $\mathcal{S} \rightarrow \mathcal{A}$, known as the optimal policy, determining the optimal decision rule for a given Markov process.

D. Update in Q-learning

A Markov policy is any mapping π_t defined over $\mathcal{S} \times \mathcal{A}$ generating an action process $\{a_t\}$ such that $\pi_t\left(s_t, a_{t+1}\right) = \Pr\left(a_{t+1}|s_t\right)$. A policy π_t is stationary if it does not depend on t and deterministic if it assigns probability 1 to a single action is each state. Notice that the optimal policy can be obtained from Q^{*} by an iterative method such as fixed point iteration [32]. However, it has two requirements. First, the transition probabilities should be known. Second, for large number of states and actions, Q^{*} is a huge sized matrix, which has large storage and computation requirements. To solve the first problem, Watkins [33] proposed Q-learning algorithm, which proceeds as follows. Consider the Markov process tuple $\mathcal{M} = (\mathcal{S}, \mathcal{A}, P_T, r, \gamma)$ and let $\{s_t\}$ be an infinite sample trajectory of the underlying Markov chain obtained with a policy π_t, yielding actions $\{a_t\}$ and rewards $\{r_t\}$. Given any initial estimate Q_0, Q-learning successively updates this estimate using the rule

$$Q_{t+1}\left(s_t, a_{t+1}\right) = Q_t\left(s_t, a_{t+1}\right) + \beta_t \Delta_t, \tag{30}$$

where $\{\beta_t\}$ is a step-size sequence and Δ_t is the temporal difference at time t,

$$\Delta_t = \left[r_t + \gamma \max_{a'} Q_t\left(s_{t+1}, a'\right)\right] - \gamma Q_t\left(s_t, a_{t+1}\right), \tag{31}$$

with $r_t = r(s_t, a_{t+1}, s_{t+1})$ being the instantaneous reward in time slot t.

If both $\mathcal{S}$ and $\mathcal{A}$ are finite sets, each estimate Q_t is simply $|\mathcal{S}| \times |\mathcal{A}|$ matrix. In that case, the convergence of Q-learning and several other related algorithms has been thoroughly studied in [34]. However, if either $\mathcal{S}$ or $\mathcal{A}$ are infinite or very large sets, explicitly representing each element of Q_t becomes infeasible to compute, update and store, and thus, some form of compact representation is needed. In this work, we present the function approximation with Q-learning, which also attains convergence.

V. Q-Learning With Function Approximation

In this section, first Q-function is linearly approximated and then non-linear based approximation is presented, along with the corresponding convergence analysis.

A. Linear function approximation (LFA)

Linear function approximation is a popular method for making Q-learning applicable to real-world settings [26]. A linear approximation in our setup is inspired by the additive form of the instantaneous costs in the ASP approximation in (16). Specifically, we propose to approximate instantaneous $Q(s_t, a_{t+1})$ to $Q_\theta(s_t, a_{t+1})$ in the time slot $t+1$ as

$$Q_\theta(s_t, a_{t+1}) = \theta_1 (1 - 2A + 2B) - 2\theta_2 B \sum_f \frac{p_{f,t}}{q_{f,t+1}}$$

$$- \theta_3 T_{ok} \frac{\Gamma(1+\alpha/2)}{(0.5C)^{\alpha/2}} \bar{\sigma}^2 \sum_f p_{f,t} q_{f,t+1}^{-\alpha/2} - \theta_4 \nu \mathbf{q}_{t+1}^T (1 - \mathbf{q}_t) \tag{32}$$

$$= \sum_{i=1}^{4} u_i (s_t, a_{t+1}) \theta_i = \mathbf{u}^T (s_t, a_{t+1}) \boldsymbol{\theta}. \tag{33}$$

where $\boldsymbol{\theta}^T = [\theta_1, \ldots, \theta_4]$ are the coefficients and $\mathbf{u}^T = [u_1, \ldots, u_4]$. Note that four linear coefficients comes from the fact that there are four terms in the ASP expression in (16).

Similar to Q-updates, in the approximation settings, the underlying idea is to apply the gradient descent to obtain the update rule for the approximated Q-learning

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + \alpha_t \nabla_\theta Q_\theta (s_t, a_{t+1}) \Delta_t \tag{34}$$

$$= \boldsymbol{\theta}_t + \alpha_t \mathbf{u} (s_t, a_{t+1}) \Delta_t, \tag{35}$$

where Δ_t is the same temporal difference as defined in (31) with Q replaced by Q_θ according to (32).

To establish the convergence of the algorithm (35), the arguments based on an ODE (ordinary differential equation) is adopted [35], establishing the trajectories of the algorithm to closely follow those of an associated ODE with a globally asymptotically stable equilibrium point.

B. Convergence of Q-learning with LFA

Here, we identify the conditions that ensure the convergence of Q-learning with linear function approximation as described in (35). To proceed, we first provide some definitions.

Given an MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, P_T, r, \gamma)$ with compact state space $\mathcal{S} \subset \mathbb{R}^{2F}$. Let $(\mathcal{S}, P_\pi)$ be the Markov chain induced by a fixed policy π. We assume the chain $(\mathcal{S}, P_\pi)$ to be uniformly ergodic with invariant probability measure μ_S and policy π to verify $\pi(s,a) > 0, \forall a \in \mathcal{A}$ and μ_S-almost all $s \in \mathcal{S}$. We denote μ_π as the probability measure defined for each measurable set $S \subset \mathcal{S}$ and each action $a \in \mathcal{A}$ as

$$\mu_\pi(S \times \{a\}) = \int_S \pi(s,a)\, \mu_S(ds). \tag{36}$$

Since the functions u_i are bounded and linearly independent, we define the covariance matrix

$$\Sigma_\pi = \mathbb{E}_\pi\left\{\mathbf{u}(s,a)\,\mathbf{u}^T(s,a)\right\} = \int_{\mathcal{S}\times\mathcal{A}} \mathbf{u}\mathbf{u}^T d\mu_\pi. \tag{37}$$

For fixed $\boldsymbol{\theta}$ and s, let the set of maximizing actions be denoted as

$$\mathcal{A}_{s,\theta} = \left\{a_\theta^* \middle| \boldsymbol{\theta}^T \mathbf{u}(s, a_\theta^*) = \max_a \boldsymbol{\theta}^T \mathbf{u}(s,a)\right\}, \tag{38}$$

which is also called the set of greedy actions. The corresponding θ-dependent covariance matrix can be written as

$$\Sigma_\theta^* = \mathbb{E}_\pi\left\{\mathbf{u}(s, a_\theta^*)\,\mathbf{u}^T(s, a_\theta^*)\right\}. \tag{39}$$

It can be noted that the difference between Σ_π and Σ_θ^* is that the former selects actions according to π, while the latter select greedy policy depending on θ. With that, the convergence result is stated in the following.

Lemma 7. *Given $\mathcal{M}$, π and $\mathbf{u}$ with finite state space, if $\forall\boldsymbol{\theta}$, $\Sigma_\pi \succeq \gamma^2\Sigma_\theta^*$ and the step size sequence verifies $\sum_t \beta_t = \infty$ and $\sum_t \beta_t^2 < \infty$, then the algorithm (35) based on linear approximation converges w.p.* 1.

Proof: Proof is based on a standard ODE argument [36, Th. 17] and can be found in [35]. ■

To satisfy the condition on the step size sequence, β_t is updated as $\beta_t = \beta_{t-1}(1 - \epsilon_\beta)$, where $\epsilon_\beta < 1$ is the decay factor, as presented in the Algorithm 1. The condition $\Sigma_\pi \succeq \gamma^2\Sigma_\theta^*$ is quite restrictive, especially when γ is close to 1. This condition essentially requires that for every $\boldsymbol{\theta}$ and for state s, we should have

$$\max_{a\in\mathcal{A}} \mathbf{u}^T(s,a)\,\boldsymbol{\theta} \approx \mathbb{E}_\pi\left\{\mathbf{u}^T(s,a)\,\boldsymbol{\theta}\right\}.$$

Therefore, such condition implies that the learning policy π is close to the policy that the algorithm is meant to compute. In other words, the maximization above yields a policy close to the policy used during learning. To satisfy this condition, the authors in [35] update the policy for taking actions at every iteration. In the proposed approach, we explore and update policy based on ϵ-greedy actions, where the exploration factor is updated at each epsiode, in turn updating the policy. The exploration factor is updated as $\epsilon_t = \epsilon_{t-1}(1-\epsilon_\delta)$, where ϵ_δ is the exploration decay rate. It means as the iterations progress, the policy is close to optimum and less exploration is needed.

Algorithm 1 Conventional, LFA and NLFA based Q-learning algorithms.

Input: $Q(s,a)=0, \forall(s,a)$, β_0, ϵ_0

Output: $Q^*(s,a)$ for optimum policy.

1: **for** $e = 1, 2, \ldots,$ max_episodes **do**

2: **for** $t = 1, 2, \ldots,$ max_steps **do**

3: Take ϵ-greedy action selection:

$$a_{t+1} = \begin{cases} U\{1, \ldots, |\mathcal{A}|\}, & U(0,1) < \epsilon_t, \\ \arg\max_b Q(s_t, b), & \text{otherwise.} \end{cases}$$

4: Observe next state $s_{t+1} = (\mathbf{p}_{t+1}, \mathbf{q}_{t+1})$.

5: Obtain reward $r_t = r(s_t, a_{t+1}, s_{t+1})$.

6: For Q-learning, update Q-values via (30).

7: For Q-learning with LFA, update θ's by (35).

8: For Q-learning with NLFA, update via (41).

9: **end for**

10: Update $\epsilon_t = \epsilon_{t-1}(1-\epsilon_\delta)$ and $\beta_t = \beta_{t-1}(1-\epsilon_\beta)$.

11: **end for**

C. Non-linear function approximation (NLFA)

Although LFA is popular, many real world applications cannot be modeled with linear functions. Here, inspired by the ASP for interference limited system in (18), we propose to approx-

imate the instantaneous $Q(s_t, a_{t+1})$ to $Q_\theta(s_t, a_{t+1})$ in time slot $t+1$ as

$$Q_\theta(s_t, a_{t+1}) = \sum_{f\in\mathcal{F}} p_{f,t}\Bigg[\frac{\theta_1 q_{f,t+1}}{2B + q_{f,t+1}\left(2A-2B+1\right)\theta_2} - \theta_3 \nu q_{f,t+1}\left(1-q_{f,t}\right)\Bigg], \tag{40}$$

where $\boldsymbol{\theta}^T = [\theta_1, \ldots, \theta_3]$ are the coefficients. Each of the coefficient is associated with the action $q_{f,t+1}$. In these settings, the update rule for the NLFA approximated Q-learning is changed to

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + \alpha_t \nabla_\theta Q_\theta\left(s_t, a_{t+1}\right)\Delta_t \tag{41}$$

where Δ_t is the same temporal difference as defined in (31) according to (32). The gradient can be calculated as

$$\begin{aligned}
\frac{\partial Q_\theta}{\partial \theta_1} &= \sum_{f\in\mathcal{F}} p_{f,t+1}\frac{q_{f,t+1}}{2B + q_{f,t+1}\left(2A-2B+1\right)\theta_2} \\
\frac{\partial Q_\theta}{\partial \theta_2} &= -\sum_{f\in\mathcal{F}} p_{f,t+1}\frac{\theta_1 q_{f,t+1}^2\left(2A-2B+1\right)}{\left(2B + q_{f,t+1}\left(2A-2B+1\right)\theta_2\right)^2} \\
\frac{\partial Q_\theta}{\partial \theta_3} &= -\nu\sum_{f\in\mathcal{F}} p_{f,t+1} q_{f,t+1}\left(1-q_{f,t}\right).
\end{aligned}$$

To establish the convergence of the algorithm (41), a similar type of approach is adopted as in [35], establishing the trajectories of the algorithm to closely follow those of an associated ODE with a globally asymptotically stable equilibrium point. This convergence result is given as follows.

Lemma 8. *Given $\mathcal{M}$, π and $\{A, B\}$ with finite state space, if the step size sequence verifies $\sum_t \beta_t = \infty$, $\sum_t \beta_t^2 < \infty$, and the following conditions are satisfied*

$$\mathbb{E}_\pi\left\{b_1\left(b_1^*\gamma - b_1\right)\right\} < 0, \tag{42}$$

$$\begin{aligned}
&\mathbb{E}_\pi\left\{b_1\left(b_1^*\gamma - b_1\right)\right\}\cdot\mathbb{E}_\pi\left\{b_3\left(b_3^*\gamma - b_3\right)\right\} \\
-&\mathbb{E}_\pi\left\{b_1\left(b_3^*\gamma - b_3\right)\right\}\cdot\mathbb{E}_\pi\left\{b_3\left(b_1^*\gamma - b_1\right)\right\} < 0,
\end{aligned} \tag{43}$$

$$\mathbb{E}_\pi\left\{b_1\left(c^*\gamma - c\right)\right\} < 0, \tag{44}$$

$$\mathbb{E}_\pi\left\{b_3\left(c^*\gamma - c\right)\right\} < 0, \tag{45}$$

where[4] $b_1 = \frac{1}{2B}\sum_{f\in\mathcal{F}} p_f q_f$, $b_3 = -\nu \sum_{f\in\mathcal{F}} p_f q_f \left(1 - q'_f\right)$, $c = -\sum_{f\in\mathcal{F}} p_f q_f^2 \frac{2A-2B+1}{4B^2}$, *then the algorithm (41) based on non-linear approximation converges w.p.* 1.

Proof: Proof is given in Appendix-C. ■

The above conditions essentially implies that at convergence, the variables b_1, b_3, c should be closed to the corresponding optimum values. Similar to the case in LFA, these conditions are satisfied by properly choosing γ and the exploration factors ϵ_β and ϵ_δ as mentioned in the Algorithm 1.

VI. Simulation Results

In simulations for ASP, we assume $M = 256$ antennas for channel hardening, $q_f = 0.2$ for moderately popular files, $\lambda_{BS} = 20$, $c_T = 0.6$, SNR $\bar{\sigma}^{-2} = 30$dB and $\alpha = 3$. Each of Q-learning algorithms is run for interference-limited systems with threshold value fixed to 0.01. Q-learning algorithm is run for finite states finite policies (FSFP) scenarios with the parameters given as follows: number of popularity profiles in the finite set $\{\mathbf{p} \in \mathcal{P}\}$, $|\mathcal{P}| = 8$, the cardinality of the set of caching probabilities $|\mathcal{A}| = 32$, content library size $F = 1024$, cache size $L = 32$, decay factor $\epsilon_\beta = 0.1$, learning rate $\beta_1 = 0.7$, the number of steps per episode is 10^3 and maximum number of episodes is 100. Note that each algorithm is initialized with the same random seed. Since the refresh rate is much higher that the ASP, we choose $\nu = 0.005$.

A. ASP versus threshold and BS's density

Figure 3 shows the variations of the success probability with the threshold for a fixed value of λ_{BS} in (a), and for different values of λ_{BS} in (b). It can be observed that as the threshold is increased, the success probability decreases, while verifying the theoretical results. As compared to MRT, ZF provides better ASP at all thresholds, although the difference in ASP is negligible at lower thresholds. From the second sub-figure, it can be seen that the success probability is independent of the density of BSs. It happens due to the fact that both the signal and interference powers are increased for an increase in the density of BSs, causing very small changes in SINRs and keeping the ASP unchanged with respect to λ_{BS}.

[4]Also note that the variables b_1, b_3, c are a function of (s, a). To avoid cumbersome notations, these notations have been shortened.

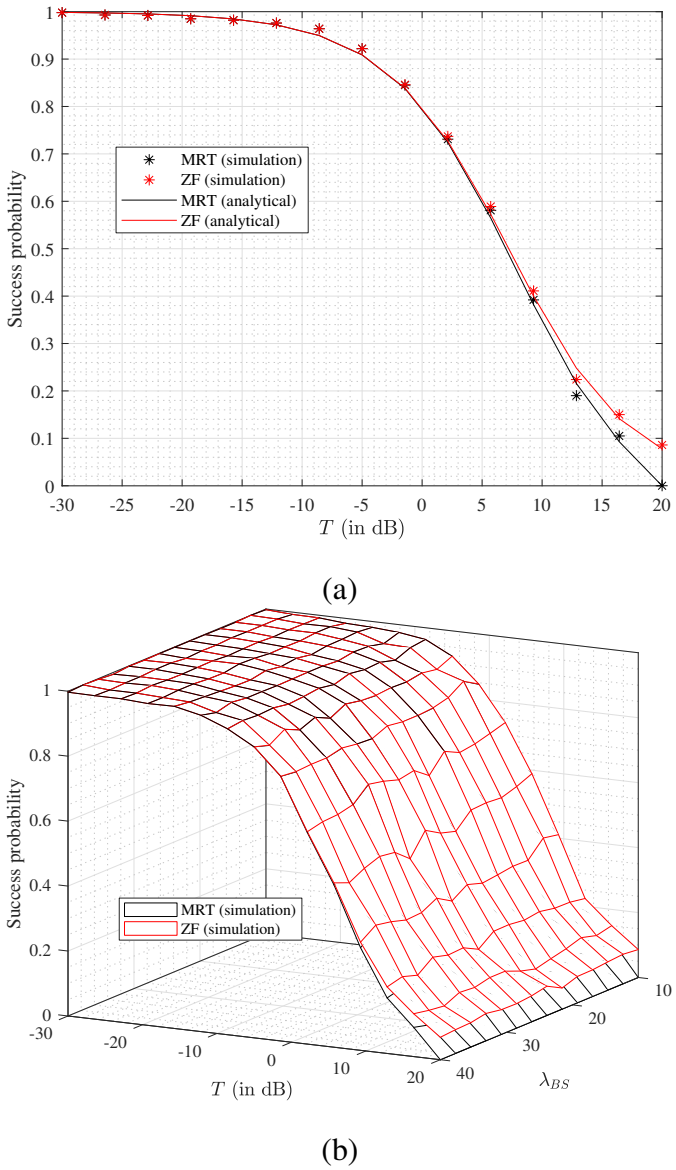


Figure 3. Success probability versus T for a fixed value $\lambda_{BS} = 20$ in (a), and for different value of λ_{BS} in (b).

B. Convergence plots

1) With single Q-entry updation: Figure 4(a) plots the episodic progress of average rewards (on the left axis) and average ASPs (on the right axis) for three different approaches, namely conventional Q-learning, and the Q-learning with LFA and NLFA with single entry updates as in conventional Q-learning. The approach with LFA (or NLFA) with single entry update is like approximated conventional Q-learning, where the Q-matrix is maintained and one entry is

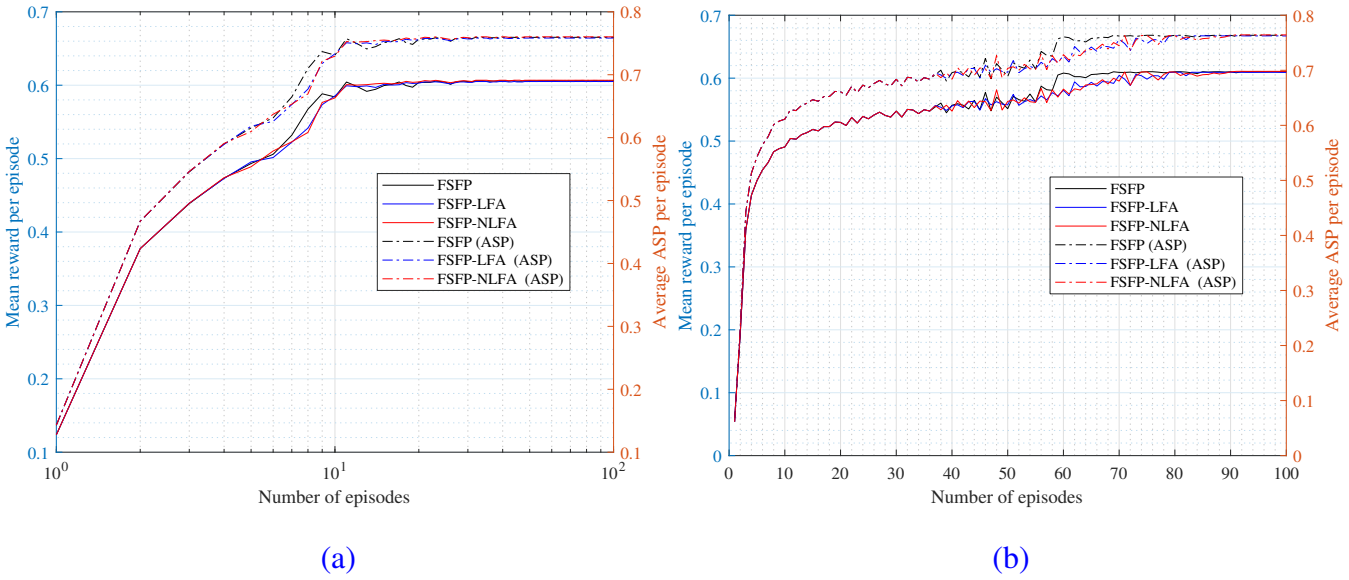


Figure 4. Convergence of average reward and average ASP versus number of episodes for single-entry update based RL algorithms for a library size $F = 1024$ and cache size $L = 32$ with 256 states and 32 actions in (a), and with 1024 states and 64 actions in (b).

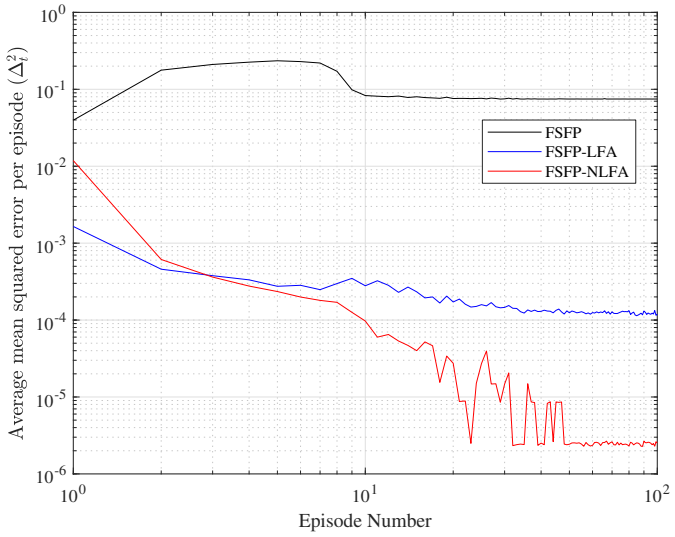
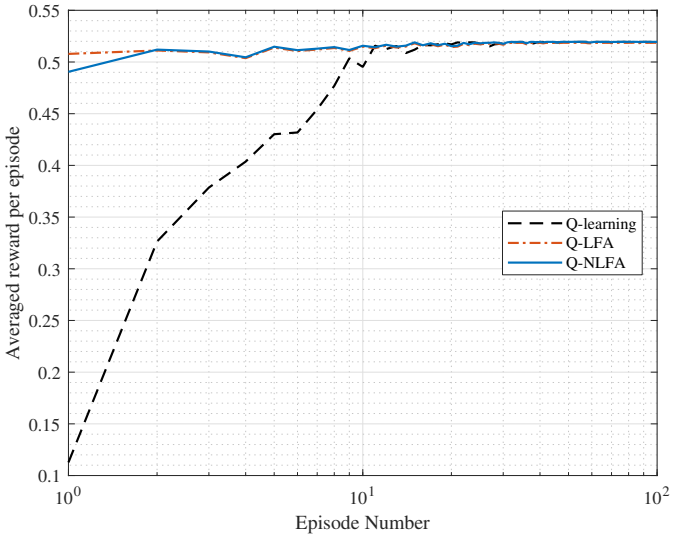


Figure 5. Convergence of average reward and average ASP versus number of episodes for single-entry update based RL algorithms for a library size $F = 1024$ and cache size $L = 32$ with 256 states and 32 actions.

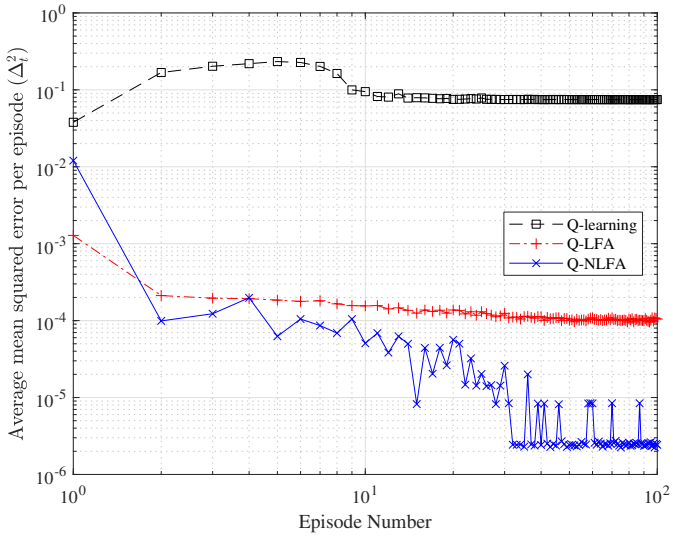
updated per step observation. It can be observed that as episodes progress, these Q-learning approaches achieve convergence around 40 episodes, and the achieved stable point is approximately same along with the approximately similar convergence path. A similar trend is observed as in Figure 4(b) for the larger set of states and actions with $|\mathcal{P}| = 16$ and $|\mathcal{A}| = 64$. These results demonstrate successful applicability of the proposed linear and non-linear approximations for the present scenario with finite set of states and actions.

Figure 5 shows the plot for the averaged mean squared error (Δ_t^2) for the same algorithms with single entry update. It can be observed that Q-values reaches a better convergence when the approximation is used. This is due to the fact that the θ-variables are updated in each step, in contrast to the occasional entry updates in the conventional Q-learning. In other words, each single Q-value is updated based on all the previous experiences of other Q-values, since θ-variables are common for all the states and actions. Next, we show the results, when all the necessary required entries are updated in each step.

2) With all necessary Q-entries updates : Figure 6(a) plots progress of the averaged reward per episode, while the averaged mean squared error (Δ_t^2) with respect the number of episodes is given in (b). As compared to single entry Q-update, the first notable fact observed from (a) is that LFA/NLFA starts giving better rewards, just after few episodes, while Q-learning needs many observation with many episodes. Note that the whole Q-matrix needn't require updation in each step; only the necessary Q-values can be computed by the updated θ-variables for the action selection by the agent in the next step. Further, it can be seen that all the three methods show convergent behavior, and the NLFA-based Q-learning provides the best value. Also, the LFA-based learning also provides better performance than the conventional Q-learning. The reason behind is the number of variables that needs to be updated or learned. In the conventional RL, the number of learning variables are huge, while in function approximated ones, the number of these variables is small. Further, better performance of NLFA than that of LFA is observed due to the fact that NLFA is able to better approximate the ASP than the linear expression in LFA. Although the converged mean squared errors for three methods are different, Figures 4 demonstrate the approximately similar reward performances, which is due to the fact that the actions depends on the relative Q-values, rather than the individual values. Thus, if the order of some of those values matches, approximately similar results can be obtained.



(a)



(b)

Figure 6. Progress of average reward per episode versus the number of steps in (a) and average mean squared error versus episodes in (b) for whole Q-update based RL algorithms for $F = 1024$ and $L = 32$ with 256 states and 32 actions.

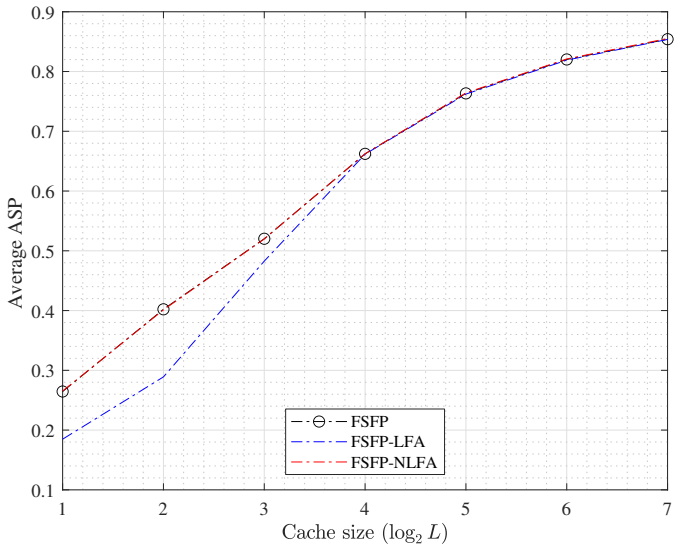


Figure 7. Average ASP with respect to cache size for $F = 1024$ with 256 states and 32 actions.

C. ASP versus cache size

Figure 7 depicts the average ASP at convergence with respect to the cache size, while keeping library size fixed. It can be seen that as the cache storage is increased, the ASP improves. For lower cache sizes, NLFA provides the better ASP than that of LFA, which reflects the limitations of LFA.

VII. Conclusion

In this paper, for a PPP network with massive-MIMO base stations, two function approximation based reinforcement learning approaches are proposed for ASP maximization and cache refresh rate minimization. We first derive the ASP for multi-user massive MIMO systems and conclude that for interference limited systems, the resulting ASP gets independent of the densities. Further, global content popularities are modeled using a Markov chain. Given a set of caching probabilities, conventional Q-learning and function approximation based Q-learning methods converge and yield the similar optimal content placement policy. The function approximated learning requires only a constant number of parameters to be updated, while the recent Q-learning model [26] in caching context requiring parameters updates proportional to number of states and actions. Using the synthetic dataset, simulations verify the equivalent performance of the approximated

Q-learning approaches with the Q-learning approach without approximation, albeit with much lower time and space computational complexity.

APPENDICES

A. *Proof of Lemma 2*

Since SINR model of interest is a function of the distance between the typical and the BS only, but not a function of the azimuth angles. Thus, Φ_{BS} is statistically equivalent to another 1D-inhomogeneous PPP $\Phi_{eq1} = \{r_i, i \in \mathbb{N}\}$ with density function $\lambda_{eq1}(r) = \int_0^{2\pi} \lambda_{BS} r d\theta = 2\pi\lambda_{BS} r$. The SINR model for Φ_{eq1} is the same as the one for Φ_{BS}. Let us define $R^{-\alpha} = \xi d^{-\alpha} \in \Phi_{eq}$. Then, $d^{\alpha} = \xi R^{\alpha}$ and $d^{\alpha-1}\mathrm{d}d = \xi R^{\alpha-1}\mathrm{d}R \implies \mathrm{d}d = dR^{-1}\mathrm{d}R = \xi^{1/\alpha}\mathrm{d}R$. It can be equated to another PPP Φ_{eq} as

$$\mathbb{E}_{\Phi_{eq1}}\left[\sum_{r\in\Phi_{eq1}} 1_{[\underline{r},\bar{r}]}(r)\right] = \mathbb{E}_{\xi,\Phi_{eq}}\left[\sum_{d\in\Phi_{eq}} 1_{[\underline{d},\bar{d}]}(d)\right]$$

$$\int_{\underline{r}}^{\bar{r}} \lambda_{eq1}(r)\mathrm{d}r = \mathbb{E}_{\xi}\left[\int_{\underline{d}}^{\bar{d}} \lambda_{eq}(d)\mathrm{d}d\right]$$

$$= \mathbb{E}_{\xi}\left[\int_{\underline{r}}^{\bar{r}} \lambda_{eq}(\xi^{1/\alpha}r)\xi^{1/\alpha}\mathrm{d}r\right]$$

$$= \int_{\underline{r}}^{\bar{r}}\left[\int_0^{\infty} a\left(\xi^{1/\alpha}r\right)^b \xi^{1/\alpha}\exp(-\xi)\mathrm{d}\xi\right]\mathrm{d}r$$

$$= a\int_{\underline{r}}^{\bar{r}} r^b\left[\int_0^{\infty} \xi^{(b+1)/\alpha}\exp(-\xi)\mathrm{d}\xi\right]\mathrm{d}r$$

$$\int_{\underline{r}}^{\bar{r}} 2\pi\lambda_{BS} r\mathrm{d}r = a\Gamma\left(1+\frac{b+1}{\alpha}\right)\int_{\underline{r}}^{\bar{r}} r^b\mathrm{d}r,$$

where $\lambda_{eq}(d) = Cd^b$ is assumed. Equating both sides yields $b = 1$ and $C = \frac{2\pi\lambda_{BS}}{\Gamma\left(1+\frac{2}{\alpha}\right)} = \frac{\pi\lambda_{BS}\alpha}{\Gamma\left(\frac{2}{\alpha}\right)}$.

B. Proof of Theorem 7

Based on the statistical equivalence, the success probability can be computed for $T = 2^{\frac{R_0}{W}} - 1$ as

$$g(q_f) = \Pr\left(\Gamma_{fk}^{MRT} > T\right) = \Pr\left(\Gamma_{fk,eq}^{MRT} > T\right) \tag{46}$$

$$= \Pr\left[\frac{\xi_{ok} d_{ok}^{-\alpha} \frac{p_{ok}}{P_T} M}{\xi_{ok} d_{ok}^{-\alpha} \left(1 - \frac{p_{ok}}{P_T}\right) + \underline{I}_{fk} + \underline{I}_f^c + \bar{\sigma}^2} > T\right] \tag{47}$$

$$= \mathbb{E}_{\xi,\Phi_{eq}}\left[\Pr\left[\xi_{ok} > T_{ok} d_{ok}^{\alpha}\left(\underline{I}_{fk} + \underline{I}_f^c + \bar{\sigma}^2\right) \middle| d_{ok}\right]\right] \tag{48}$$

$$= \mathbb{E}_{\xi,\Phi_{eq}}\left[\exp\left(-T_{ok} d_{ok}^{\alpha}\left(\underline{I}_{fk} + \underline{I}_f^c + \bar{\sigma}^2\right)\right)\right], \tag{49}$$

where the last step is obtained from the exponential distribution of ξ_{ok}, and $T_{ok} = \frac{T}{M\frac{p_{ok}}{P_T} - T\left(1 - \frac{p_{ok}}{P_T}\right)} = \frac{TP_T}{Mp_{ok}} \cdot \frac{1}{1 - \frac{T}{M}\left(\frac{P_T}{p_{ok}} - 1\right)}$. For equal power allocation $p_{ok} = \frac{P_T}{K_k}$, the effective threshold $T_{ok} = \frac{TK_k}{M} \cdot \frac{1}{1 - \frac{T}{M}(K_k - 1)}$ changes with the number of users. On the other hand, if per user power allocation is constant $p_{ok} = c_T P_T$ i.e. the total transmit power budget is increased with the increase in number of users, $T_{ok} = \frac{T}{Mc_T} \cdot \frac{1}{1 - \frac{T}{M}\left(c_T^{-1} - 1\right)} = \bar{T}$ is a constant. For ZF, $T_{ok} = \frac{TP_T}{Mp_{ok}}$.

Further, since the terms in $\underline{I}_{fk} + \underline{I}_f^c$ are independent, the first term in the above equation can be simplified as

$$\mathbb{E}_{\xi,\Phi_{eq}}\left\{\exp\left(-T_{ok} d_{ok}^{\alpha} \underline{I}_{fk}\right)\right\} \tag{50}$$

$$= \mathbb{E}_{\Phi_{eq}}\left\{\prod_{j \in \Phi_{eq}(f)\setminus\{k\}} \mathbb{E}_{\xi}\left\{\exp\left(-\xi d_{oj}^{-\alpha} T_{ok} d_{ok}^{\alpha}\right)\right\}\right\} \tag{51}$$

$$= \mathbb{E}_{\Phi_{eq}}\left[\prod_{j \in \Phi_{eq}(f)\setminus\{k\}} \frac{1}{1 + d_{oj}^{-\alpha} T_{ok} d_{ok}^{\alpha}}\right] \tag{52}$$

$$\overset{(a)}{=} \exp\left[-\int_0^{\infty}\left(1 - \frac{1}{1 + d^{-\alpha} T_{ok} d_{ok}^{\alpha}}\right) q_f \lambda_{eq}(d) \mathrm{d}d\right]$$

$$= \exp\left[-\int_0^{\infty}\left(\frac{q_f C d}{1 + d^{\alpha} T_{ok}^{-1} d_{ok}^{-\alpha}}\right) \mathrm{d}d\right] \tag{53}$$

$$\overset{(b)}{=} \exp\left[-C q_f \alpha^{-1} T_{ok}^{2/\alpha} d_{ok}^2 \int_0^{\infty} \frac{t^{2/\alpha - 1} \mathrm{d}t}{1 + t}\right] \tag{54}$$

$$\overset{(c)}{=} \exp\left[-C q_f A d_{ok}^2\right] \tag{55}$$

where in (a), Campbell's theorem is invoked; in (b), by change of variables $t = d^{\alpha} T_{ok}^{-1} d_{ok}^{-\alpha}$, we get $\mathrm{d}t = \mathrm{d}d \cdot \alpha d^{\alpha-1} T_{ok}^{-1} d_{ok}^{-\alpha} = \mathrm{d}d \cdot \alpha d^{-1} t = \mathrm{d}d \cdot \alpha t \left(T_{ok}^{-1} d_{ok}^{-\alpha} t^{-1}\right)^{1/\alpha}$ and $d \cdot \mathrm{d}d = \alpha^{-1} d^2 t^{-1} \mathrm{d}t = \alpha^{-1} T_{ok}^{2/\alpha} d_{ok}^2 t^{2/\alpha - 1} \mathrm{d}t$; (c) follows from letting $A = \alpha^{-1} T_{ok}^{2/\alpha} I(0)$ and $I(x) = \int_x^{\infty} \frac{c^{2/\alpha - 1} \mathrm{d}c}{1+c}$. Similarly, the other term of (49) can be written as

$$\begin{aligned} &\mathbb{E}_{\xi, \Phi_{eq}} \left\{ \exp\left(-T_{ok} d_{ok}^{\alpha} \underline{I}_f^c\right) \right\} \\ &= \exp\left[-(1-q_f) C B d_{ok}^2\right], \end{aligned} \tag{56}$$

where $B = \alpha^{-1} T_{ok}^{2/\alpha} I(T_{ok}^{-1})$. Substituting (55) and (56) into (49) gives the required expression in (11).

C. *Proof of Lemma 8*

To prove the convergence of non-linear function approximation based Q-learning algorithm, we leverage the Taylor series approximation for Q-function as

$$Q_{\theta}(s,a) \approx \mathbf{b}^T \boldsymbol{\theta} + \boldsymbol{\theta}^T \mathbf{C} \boldsymbol{\theta},$$

where $\mathbf{b}^T = \nabla_{\theta}^T Q_{\theta}\Big|_{\theta=0} = [b_1, 0, b_3] = \left[\frac{1}{2B} \sum_{f \in \mathcal{F}} p_f q_f, 0, -\nu \sum_{f \in \mathcal{F}} p_f q_f \left(1 - q_f'\right)\right]$ and $\mathbf{C} = \nabla_{\theta} \nabla_{\theta}^T Q_{\theta}\Big|_{\theta=0} = \begin{bmatrix} 0 & c & 0 \\ c & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ with $c = -\sum_{f \in \mathcal{F}} p_f q_f^2 \frac{2A - 2B + 1}{4B^2}$. Let $\max_a Q(s,a) = \mathbf{b}_{\theta}^T \boldsymbol{\theta} + \boldsymbol{\theta}^T \mathbf{C}_{\theta} \boldsymbol{\theta}$. The proof is established via a standard ODE argument. The assumptions on the chain $(\mathcal{M}, \pi)$, the function u and μ_X-almost every $x \in \mathcal{X}$ ensure the applicability of the result in [36, Th. 17, p239]. Therefore, the convergence of the algorithm can be analyzed in terms of the stability of the equilibrium points of the associated ODE

$$\dot{\boldsymbol{\theta}} = \mathbb{E}_{\pi} \left\{ \nabla_{\theta} \mathcal{E} \right\}, \tag{57}$$

where $\mathcal{E} = \left(r(s,a,s') + \gamma \max_b Q_{\theta}(s,b) - Q_{\theta}(s,a)\right)^2$ and

$$\begin{aligned} \mathbb{E}_{\pi} \left\{ \nabla_{\theta} \mathcal{E} \right\} &= \mathbb{E}_{\pi} \left\{ \frac{\partial \mathcal{E}}{\partial Q_{\theta}} \nabla_{\theta} Q_{\theta} \right\} \\ &= \mathbb{E}_{\pi} \Big\{ \left(r(s,a,s') + \gamma \left(\mathbf{b}_{\theta}^T \boldsymbol{\theta} + \boldsymbol{\theta}^T \mathbf{C}_{\theta} \boldsymbol{\theta}\right) \right. \\ &\quad \left. - \left(\mathbf{b}^T \boldsymbol{\theta} + \boldsymbol{\theta}^T \mathbf{C} \boldsymbol{\theta}\right) \right) \left(\mathbf{b} + \mathbf{C} \boldsymbol{\theta}\right) \Big\}. \end{aligned}$$

If the ODE in (57) has a global asymptotically stable point, the algorithm $\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + \beta_t \nabla_\theta \mathcal{E}$ converges w.p. 1 [36]. Let $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$ be two trajectories of ODE starting at different initializations, and let $\tilde{\boldsymbol{\theta}} = \boldsymbol{\theta}_1 - \boldsymbol{\theta}_2$. From (57), we get

$$\begin{aligned}
\frac{\partial}{\partial t}\|\tilde{\boldsymbol{\theta}}\|_2^2 &= 2\tilde{\boldsymbol{\theta}}^T \left(\dot{\boldsymbol{\theta}}_1 - \dot{\boldsymbol{\theta}}_2\right) \\
&= 2\tilde{\boldsymbol{\theta}}^T \mathbb{E}_\pi \Big\{ \gamma \left(\mathbf{b} + \mathbf{C}\boldsymbol{\theta}_1\right) \left(\mathbf{b}_{\theta_1} + \mathbf{C}_{\theta_1}\boldsymbol{\theta}_1\right)^T \boldsymbol{\theta}_1 \\
&\quad - \left(\mathbf{b} + \mathbf{C}\boldsymbol{\theta}_1\right) \left(\mathbf{b} + \mathbf{C}\boldsymbol{\theta}_1\right)^T \boldsymbol{\theta}_1 \\
&\quad - \gamma \left(\mathbf{b} + \mathbf{C}\boldsymbol{\theta}_2\right) \left(\mathbf{b}_{\theta_2} + \mathbf{C}_{\theta_2}\boldsymbol{\theta}_2\right)^T \boldsymbol{\theta}_2 \\
&\quad + \left(\mathbf{b} + \mathbf{C}\boldsymbol{\theta}_2\right) \left(\mathbf{b} + \mathbf{C}\boldsymbol{\theta}_2\right)^T \boldsymbol{\theta}_2 \Big\}.
\end{aligned}$$

To get $\frac{\partial}{\partial t}\|\tilde{\boldsymbol{\theta}}\|_2^2 < 0$, we need to have the following inequalities as

$$\gamma \tilde{\boldsymbol{\theta}}^T \mathbb{E}_\pi \left\{ \mathbf{b} \left(\mathbf{b}_{\theta_1}^T \boldsymbol{\theta}_1 - \mathbf{b}_{\theta_2}^T \boldsymbol{\theta}_2 \right) \right\} < \tilde{\boldsymbol{\theta}}^T \mathbb{E}_\pi \left\{ \mathbf{b}\mathbf{b}^T \right\} \tilde{\boldsymbol{\theta}}, \tag{58}$$

$$\begin{aligned}
&\gamma \tilde{\boldsymbol{\theta}}^T \mathbb{E}_\pi \left\{ \mathbf{C}\boldsymbol{\theta}_1 \boldsymbol{\theta}_1^T \mathbf{b}_{\theta_1} - \mathbf{C}\boldsymbol{\theta}_2 \boldsymbol{\theta}_2^T \mathbf{b}_{\theta_2} \right\} \\
&< \mathbb{E}_\pi \left\{ tr\left(\mathbf{b}\tilde{\boldsymbol{\theta}}^T \mathbf{C}\boldsymbol{\theta}_1 \boldsymbol{\theta}_1^T \right) - tr\left(\mathbf{b}\tilde{\boldsymbol{\theta}}^T \mathbf{C}\boldsymbol{\theta}_2 \boldsymbol{\theta}_2^T \right) \right\},
\end{aligned} \tag{59}$$

$$\begin{aligned}
&\gamma \mathbb{E}_\pi \left\{ \tilde{\boldsymbol{\theta}}^T \mathbf{b}\, tr\left(\mathbf{C}_{\theta_1} \boldsymbol{\theta}_1 \boldsymbol{\theta}_1^T - \mathbf{C}_{\theta_2} \boldsymbol{\theta}_2 \boldsymbol{\theta}_2^T \right) \right\} \\
&< \mathbb{E}_\pi \left\{ \tilde{\boldsymbol{\theta}}^T \mathbf{b}\, tr\left(\mathbf{C}\boldsymbol{\theta}_1 \boldsymbol{\theta}_1^T - \mathbf{C}\boldsymbol{\theta}_2 \boldsymbol{\theta}_2^T \right) \right\},
\end{aligned} \tag{60}$$

$$\begin{aligned}
&\gamma \tilde{\boldsymbol{\theta}}^T \mathbb{E}_\pi \left\{ \mathbf{C}\boldsymbol{\theta}_1 \boldsymbol{\theta}_1^T \mathbf{C}_{\theta_1} \boldsymbol{\theta}_1 - \mathbf{C}\boldsymbol{\theta}_2 \boldsymbol{\theta}_2^T \mathbf{C}_{\theta_2} \boldsymbol{\theta}_2 \right\} \\
&< \tilde{\boldsymbol{\theta}}^T \mathbb{E}_\pi \left\{ \mathbf{C}\boldsymbol{\theta}_1 \boldsymbol{\theta}_1^T \mathbf{C}\boldsymbol{\theta}_1 - \mathbf{C}\boldsymbol{\theta}_2 \boldsymbol{\theta}_2^T \mathbf{C}\boldsymbol{\theta}_2 \right\}.
\end{aligned} \tag{61}$$

If $\mathbf{b}_{\theta_1}^T \boldsymbol{\theta}_2 \leq \mathbf{b}_{\theta_2}^T \boldsymbol{\theta}_2$, the first inequality in (58) reduces to $\tilde{\boldsymbol{\theta}}^T \mathbb{E}_\pi \left\{ \mathbf{b}\mathbf{b}_{\theta_1}^T \right\} \tilde{\boldsymbol{\theta}} \gamma < \tilde{\boldsymbol{\theta}}^T \mathbb{E}_\pi \left\{ \mathbf{b}\mathbf{b}^T \right\} \tilde{\boldsymbol{\theta}}$, yielding

$$\mathbb{E}_\pi \left\{ \mathbf{b}\mathbf{b}_{\theta_1}^T \right\} \gamma \prec \mathbb{E}_\pi \left\{ \mathbf{b}\mathbf{b}^T \right\}. \tag{62}$$

Similarly, the second inequality is satisfied, if above condition is followed. The third inequality is satisfied if

$$\mathbb{E}_\pi \left\{ \mathbf{b}\mathcal{V}(\mathbf{C}_{\theta_1})^T \right\} \gamma < \mathbb{E}_\pi \left\{ \mathbf{b}\mathcal{V}(\mathbf{C})^T \right\}, \tag{63}$$

where $\mathcal{V}(\cdot)$ denotes the vectorization operation. This inequality implies $\mathbf{C}_{\theta_1}\gamma \prec \mathbf{C}$ for most of (s, a) in the expectation, which leads to the forth inequality satisfied. This means, $\tilde{\boldsymbol{\theta}}$ converges asymptotically to the origin i.e. the ODE in (57) is globally asymptotically stable. Since the ODE is time-invariant, there exists one globally asymptotically stable point for the ODE.

The conditions can be simplified as follows. $\mathbb{E}_\pi\left\{\mathbf{b}\left(\mathbf{b}_{\theta_1}\gamma-\mathbf{b}\right)^T\right\}=\sum_a \pi\left(s,a\right)\mathbf{b}\left(\mathbf{b}_{\theta_1}\gamma-\mathbf{b}\right)^T$.

$$\mathbb{E}_\pi \mathbf{b}\left(\mathbf{b}_{\theta_1}\gamma-\mathbf{b}\right)^T=\mathbb{E}_\pi\begin{bmatrix} b_1 \\ 0 \\ b_3 \end{bmatrix}\begin{bmatrix} b_1^*\gamma-b_1 \\ 0 \\ b_3^*\gamma-b_3 \end{bmatrix}^T$$

$$=\begin{bmatrix} \mathbb{E}_\pi b_1\left(b_1^*\gamma-b_1\right) & 0 & \mathbb{E}_\pi b_1\left(b_3^*\gamma-b_3\right) \\ 0 & 0 & 0 \\ \mathbb{E}_\pi b_3\left(b_1^*\gamma-b_1\right) & 0 & \mathbb{E}_\pi b_3\left(b_3^*\gamma-b_3\right) \end{bmatrix}\prec 0$$

if $\mathbb{E}_\pi b_1\left(b_1^*\gamma-b_1\right)>0$ and $\mathbb{E}_\pi b_1\left(b_1^*\gamma-b_1\right)\cdot\mathbb{E}_\pi b_3\left(b_3^*\gamma-b_3\right)-\mathbb{E}_\pi b_1\left(b_3^*\gamma-b_3\right)\cdot\mathbb{E}_\pi b_3\left(b_1^*\gamma-b_1\right)<0$. Further, the second condition reduces to

$$\mathbb{E}_\pi\left\{\mathbf{b}\mathcal{V}(\mathbf{C}_{\theta_1})^T\gamma-\mathbf{b}\mathcal{V}(\mathbf{C})^T\right\}$$

$$=\begin{bmatrix} & \mathbb{E}_\pi b_1\left(c^*\gamma-c\right) & 0 & \mathbb{E}_\pi b_1\left(c^*\gamma-c\right) & \\ \mathbf{0}, & 0 & 0 & 0 & ,\mathbf{0} \\ & \mathbb{E}_\pi b_3\left(c^*\gamma-c\right) & 0 & \mathbb{E}_\pi b_3\left(c^*\gamma-c\right) & \end{bmatrix}<0,$$

which leads to $\mathbb{E}_\pi b_1\left(c^*\gamma-c\right)<0$ and $\mathbb{E}_\pi b_3\left(c^*\gamma-c\right)<0$.

ACKNOWLEDGMENT

This work was supported by the UK Engineering and Physical Sciences Research Council (EPSRC) under grants EP/P009549/1, EP/P009670/1; UK-India Education and Research Initiative Thematic Partnerships under grants DST-UKIERI-2016-17-0060, DST/INT/UK/P-129/2016, UGC-UKIERI 2016-17-058, and SPARC/2018-2019/P148/SL.

REFERENCES

[1] T. Muhammed, R. Mehmood, A. Albeshri, and I. Katib, "Ubehealth: A personalized ubiquitous cloud and edge-enabled networked healthcare system for smart cities," *IEEE Access*, vol. 6, pp. 32 258–32 285, 2018.

[2] H. S. Goian, O. Y. Al-Jarrah, S. Muhaidat, Y. Al-Hammadi, P. Yoo, and M. Dianati, "Popularity-based video caching techniques for cache-enabled networks: A survey," *IEEE Access*, vol. 7, pp. 27 699–27 719, 2019.

[3] L. Li, G. Zhao, and R. S. Blum, "A survey of caching techniques in cellular networks: Research issues and challenges in content placement and delivery strategies," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 3, pp. 1710–1732, 2018.

[4] K. Shanmugam, N. Golrezaei, A. G. Dimakis, A. F. Molisch, and G. Caire, "Femtocaching: Wireless content delivery through distributed caching helpers," *IEEE Transactions on Information Theory*, vol. 59, no. 12, pp. 8402–8413, 2013.

[5] K. Poularakis and L. Tassiulas, "On the complexity of optimal content placement in hierarchical caching networks," *IEEE Transactions on Communications*, vol. 64, no. 5, pp. 2092–2103, 2016.

[6] B. Blaszczyszyn and A. Giovanidis, "Optimal geographic caching in cellular networks," in *IEEE International Conference on Communications (ICC)*, 2015, pp. 3358–3363.

[7] B. Serbetci and J. Goseling, "Optimal geographical caching in heterogeneous cellular networks with nonhomogeneous helpers," *arXiv preprint arXiv:1710.09626*, 2017.

[8] D. Liu and C. Yang, "Caching policy toward maximal success probability and area spectral efficiency of cache-enabled HetNets," *IEEE Transactions on Communications*, vol. 65, no. 6, pp. 2699–2714, 2017.

[9] D. Liu, B. Chen, C. Yang, and A. F. Molisch, "Caching at the wireless edge: design aspects, challenges, and future directions," *IEEE Communications Magazine*, vol. 54, no. 9, pp. 22–28, 2016.

[10] K. Avrachenkov, J. Goseling, and B. Serbetci, "A low-complexity approach to distributed cooperative caching with geographic constraints," *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, vol. 1, no. 1, p. 27, 2017.

[11] K. Avrachenkov, X. Bai, and J. Goseling, "Optimization of caching devices with geometric constraints," *Performance Evaluation*, vol. 113, pp. 68–82, 2017.

[12] J. Yin, L. Li, H. Zhang, X. Li, A. Gao, and Z. Han, "A prediction-based coordination caching scheme for content centric networking," in *27th Wireless and Optical Communication Conference (WOCC)*, 2018, pp. 1–5.

[13] W.-X. Liu, J. Zhang, Z.-W. Liang, L.-X. Peng, and J. Cai, "Content popularity prediction and caching for ICN: A deep learning approach with SDN," *IEEE Access*, vol. 6, pp. 5075–5089, 2018.

[14] H. Nakayama, S. Ata, and I. Oka, "Caching algorithm for content-oriented networks using prediction of popularity of contents," in *IFIP/IEEE International Symposium on Integrated Network Management (IM)*, 2015, pp. 1171–1176.

[15] Y. Zhang, X. Tan, and W. Li, "PPC: Popularity prediction caching in ICN," *IEEE Communications Letters*, vol. 22, no. 1, pp. 5–8, 2018.

[16] B. Bharath, K. G. Nagananda, and H. V. Poor, "A learning-based approach to caching in heterogenous small cell networks," *IEEE Transactions on Communications*, vol. 64, no. 4, pp. 1674–1686, 2016.

[17] R. Haw, S. M. A. Kazmi, K. Thar, M. G. R. Alam, and C. S. Hong, "Cache aware user association for wireless heterogeneous networks," *IEEE Access*, vol. 7, pp. 3472–3485, 2019.

[18] J. Wu, Y. Zhou, D. M. Chiu, and Z. Zhu, "Modeling dynamics of online video popularity," *IEEE Transactions on Multimedia*, vol. 18, no. 9, pp. 1882–1895, Sep. 2016.

[19] Y. Jiang, M. Ma, M. Bennis, F. Zheng, and X. You, "User preference learning-based edge caching for fog radio access network," *IEEE Transactions on Communications*, vol. 67, no. 2, pp. 1268–1283, Feb 2019.

[20] R. Wang, R. Li, P. Wang, and E. Liu, "Analysis and optimization of caching in fog radio access networks," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 8, pp. 8279–8283, 2019.

[21] F. A. Khan, H. He, J. Xue, and T. Ratnarajah, "Performance analysis of cloud radio access networks with distributed multiple antenna remote radio heads," *IEEE Transactions on Signal Processing*, vol. 63, no. 18, pp. 4784–4799, 2015.

[22] G. S. Paschos, G. Iosifidis, M. Tao, D. Towsley, and G. Caire, "The role of caching in future communication systems and networks," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 6, pp. 1111–1125, 2018.

[23] S. O. Somuyiwa, D. Gündüz, and A. Gyorgy, "Reinforcement learning for proactive caching of contents with different demand probabilities," in *15th International Symposium on Wireless Communication Systems (ISWCS)*, 2018, pp. 1–6.

[24] W. Li, J. Wang, G. Zhang, L. Li, Z. Dang, and S. Li, "A reinforcement learning based smart cache strategy for cache-aided ultra-dense network," *IEEE Access*, vol. 7, pp. 39 390–39 401, 2019.

[25] N. Garg, M. Sellathurai, and T. Ratnarajah, "Content placement learning for success probability maximization in wireless edge caching networks," in *IEEE ICASSP*, May 2019, pp. 3092–3096.

[26] A. Sadeghi, F. Sheikholeslami, and G. B. Giannakis, "Optimal and scalable caching for 5G using reinforcement learning of space-time popularities," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 1, pp. 180–190, 2018.

[27] N. Garg, M. Sellathurai, V. Bhatia, B. N. Bharath, and T. Ratnarajah, "Online content popularity prediction and learning in wireless edge caching," *IEEE Transactions on Communications*, vol. 68, no. 2, pp. 1087–1100, 2020.

[28] N. Garg, A. Jain, and G. Sharma, "Partially Loaded Superimposed Training Scheme for Large MIMO Uplink Systems," *Wireless Personal Communications*, vol. 100, no. 4, pp. 1313–1338, 2018.

[29] L. Yin and H. Haas, "Coverage analysis of multiuser visible light communication networks," *IEEE Transactions on Wireless Communications*, vol. 17, no. 3, pp. 1630–1643, March 2018.

[30] T. D. Novlan, H. S. Dhillon, and J. G. Andrews, "Analytical modeling of uplink cellular networks," *IEEE Transactions on Wireless Communications*, vol. 12, no. 6, pp. 2669–2679, June 2013.

[31] S. M. Yu and S.-L. Kim, "Downlink capacity and base station density in cellular networks," in *11th international symposium and workshops on modeling and optimization in mobile, ad hoc and wireless networks (WiOpt)*. IEEE, 2013, pp. 119–124.

[32] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge University Press, 2004.

[33] C. J. C. H. Watkins and P. Dayan, "Q-learning," *Machine Learning*, 1992.

[34] J. N. Tsitsiklis and B. Van Roy, "An analysis of temporal-difference learning with function approximation," *IEEE Transactions on Automatic Control*, vol. 42, no. 5, pp. 674–690, 1997.

[35] F. S. Melo, S. P. Meyn, and M. I. Ribeiro, "An analysis of reinforcement learning with function approximation," *Proceedings of the 25th International Conference on Machine Learning*, pp. 664–671, 2008.

[36] A. Benveniste, M. Métivier, and P. Priouret, *Adaptive Algorithms and Stochastic Approximations*, 1990.